Dependency Diagrams (Python 3.7)

Dependency diagrams

"Dependency diagrams" give a visual representation of the structure of a programming project, showing how modules are connected, or how functions are connected by calls (which function calls which other functions). Dependency diagrams are also called "call graphs" or "code maps".

Reasons for creating dependency diagrams:

1. Useful for code testing and debugging
2. Useful for getting to know code written by others (including coursework submissions!)
3. More dependencies (higher complexity) may reduce code efficiency
4. More dependencies can make code harder to maintain
5. More dependencies may also have a good point, showing that code is modular.

Because of 2., you need to submit a dependency diagram for your coursework with the code.

There are various methods for creating dependency diagrams or call graphs. Some only look at module level, which is not much use where a module contains several functions. The method described below is the one we recommend because it looks at function level and provides a good balance between clarity and detail. It is also a static method, which scans the .py files without running them. The main advantage is that this could potentially also work for incomplete or buggy code.

Setting up: Required Software

Ensure Python 3.7 is installed. Next, download the following software:

- Graphviz 2.38: https://graphviz.gitlab.io/_pages/Download/windows/graphviz-2.38.msi
- Pyan3: https://github.com/davidfraser/pyan/archive/master.zip

For Graphviz, follow the standard installation wizard.

For Pyan3, extract the source code to a destination of your own choosing. Graphviz is required in order to interpret and convert the graph created by Pyan into an image format (i.e. PNG).

Usage

Make sure that the code you want to draw a diagram for is contained in a single directory (e.g. C:\coursework), and that this directory contains no other, unrelated code files.

Open the Windows Command prompt and navigate to the root directory of the Pyan3 source code which was previously downloaded. To generate a call graph, the following command is used:

```
py pyan.py "coursework\*.py" --no-defines --colored --grouped --dot >mygraph.dot
```

Red text is user specified values. In this example, we generate a call graph for all Python scripts located in a directory named coursework. You may need to specify the full path to this directory (the code above assumes it is a subdirectory of the Pyan3 directory)

The first three flags (`--no-defines`, `--colored` and `--grouped`) are used to set certain visual aspects of our graph while `--dot` tells Pyan to output the graph in dot format. Thus, the output of the above command will be a .dot graph named 'mygraph.dot'.

Once we have the .dot graph, we need to use Graphviz to convert this into an image format. The following command can be used in the same command prompt:

```
dot -Tpng mygraph.dot -o mygraph.png
```

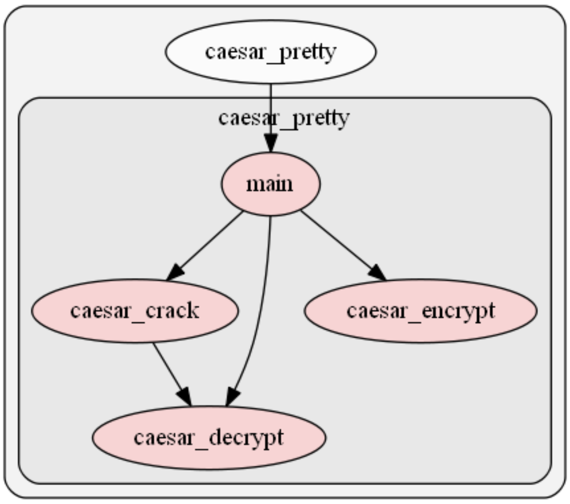
This will convert mygraph.dot to a PNG file named mygraph.png. You can then view the .png file with any image viewer. **The above command may not work** if the Graphviz path environment variables were not setup correctly during installation. In that case, locate and use the full path name to the Graphviz **dot.exe** executable instead. For example, the following command can be used:

```
"C:\Program Files (x86)\Graphviz2.38\bin\dot.exe" -Tpng mygraph.dot
-o mygraph.png
```

Assuming dot.exe is located in the `'C:\Program Files (x86)\Graphviz2.38\bin\'` directory .

Try it out!

Generate a dependency diagram for your Caesar cipher application. Mine looks like this:



Troubleshooting and Notes

- If your dot file is empty, it didn't find your script files – check the path, or copy the script files into the pyan-master directory.
- If you get an error message converting the dot file to a png, check that none of your py files have spaces in the filenames. Rename them if necessary!

The first two commands should also work for Linux and MacOS systems using the bash terminal. Just make sure to install the correct Graphviz package for your chosen operating system rather than the Windows version.